UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER OF PATENTS AND TRADEMARKS
Washington, D.C. 20231
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/212,291	12/16/1998	CHINNA PRUDVI	2207/5915	8642

23838 7590 02/10/2003

KENYON & KENYON
1500 K STREET, N.W., SUITE 700
WASHINGTON, DC 20005

EXAMINER	
THAI, TUAN V	

ART UNIT	PAPER NUMBER
2186	

DATE MAILED: 02/10/2003

Please find below and/or attached an Office communication concerning this application or proceeding.

UNITED STATES DEPARTMENT OF COMMERCE
Patent and Trademark Office
ASSISTANT SECRETARY AND COMMISSIONER OF
PATENTS AND TRADEMARKS
Washington, D.C. 20231

BEFORE THE BOARD OF PATENT APPEALS
AND INTERFERENCES

Paper No. 22

Serial Number: 09/212,291
Filing Date: December/16/98
Appellant(s): Prudvi et al.

Clyde E. Findley (Reg. No. 50,724)
For Appellant

EXAMINER'S ANSWER

I. This is in response to appellant's brief on appeal filed 11/26/02. The amendment after final filed on the same date of the appeal brief to cancel claims 8-10 and 22 has been entered and considered.

(1) *Status of claims.*

Claims 24 and 29 stand rejected under 35 U.S.C. 102(b) as anticipated by Sachs, et al. U.S. Patent No. 4,884,197 ("Sachs"); (Claims 8-10 and 22 has been canceled by Amendment filed 11/26/02; the rejections of claims 17-21, 23-24 and 27-29 as being clearly anticipated by Sachs (cited by Examiner on page 6 of the Final Office Action, paper #18) is a typographical error; however deemed to be convinced by Applicant's arguments and therefore being withdrawn by Examiner. Any inconvenience is sincerely regretted).

Claims 17-21, 23 and 27-28 stand rejected under 35 U.S.C. 102(b) as being anticipated by Scales, et al., U.S. Patent No. 4,914,573 ("Scales").

Claims 1-7, 11-16 and 25-26 stand rejected under 35 U.S.C 103(a) over Sachs in view of Scales.

(2) *Status of Amendments After Final.*

The appellant's statement of the status of amendments after final rejection contained in the brief is correct.

(3) *Summary of invention.*

The summary of invention contained in the brief is correct.

(4) *Issues.*

The appellant's statement of issues 2 and 3 in the brief is correct. Issue 1 however should be read as "Whether claims 24 and 29 patentably distinguish over Sachs under 35 U.S.C. 102(b)".

(5) *Grouping of claims.*

Appellant's brief includes statements that (a) Claims 1-7 and 25 are directed to a processing agent with an internal cache having cache entries that are sized to store multiple data line lengths of data. stand or fall together, (b) Claims 11-16, 24, 26 and 29 describe a processing agent comprising an internal cache

with cache entries sized to store multiple data lines, and a transaction queue system that posts external transaction related to a single data line, (c) Claims 17-21, 23 and 27-28 are directed to methods of processing a data request comprising posting a series of external data line transactions for a cache line when the data request misses the cache.

(6) *Claims appealed.*

The copy of the appealed claims contained in the Appendix to the brief is correct.

(7) *Prior Art of record.*

The following is a listing of the prior art of record relied upon in the rejection of claims under appeal.

(a)	4,914,573	Scales	04-1990
(B)	4,884,197	Sachs	11-1994

(8) *New prior art.*

No new prior art has been applied in this examiner's answer.

(9) *Grounds of objection/rejection.*

Claim Rejections - 35 USC § 102

1. The following is a quotation of the appropriate paragraphs

of 35 U.S.C. § 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless --
>
> (b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

2. Claims 24 and 29 are rejected under 35 U.S.C. 102(b) as being anticipated by Sachs et al. (USPN: 4,884,197) (hereinafter Sachs).

As per claim 24, Sachs teaches the invention as claimed including a processing agent (e.g. see figures 8 and 9), the agent comprising an cache memory 320 having a plurality of cache lines; for example, Sachs discloses that the cache memory 320 having three fields, a used bit field, and two identical read-write memory fields W and X wherein each field W and X contain multiple lines, and each line in the cache contains multiple entries (fields) (e.g. see figure 10.B) and having tag field RA, line valid bit LV, line dirty bit LD and multiple data entries DT (e.g. see column 22, lines 14 et seq.); Sachs further discloses that the cache entries include a tag portion for storing address information as being the real address field RA (e.g. see figure 10B, and column 22, lines 32 et seq.); wherein the processing agent posts a series of external transactions related to the address information, each of the external transactions filling one of the cache entries in the cache lines is taught by Sachs as

when the requested information for the addressed location is not stored in the respective cache-MMU memories (cache misses); the micro engine of the cache-MMUs provides a translated physical address for output to the main memory 140, the corresponding information is thereafter transferred from main memory to the respective instruction cache-MMU or to/from the cache-MMU and as needed to the processor 110 (e.g. see column 6, lines 13-21);

As per claim 29, wherein data line corresponds to the maximum amount of data that can be transferred in a single bus transaction is taught by Sachs to the extent that it is being claimed; for example, when detailing data cache bus, Sachs discloses on store operations, the CPU puts an address following by data on the address/data bus for one (single) clock/bus cycle (e.g. see column 6, lines 27 et seq.);

3. Claims 17-21, 23 and 27-28 are rejected under 35 U.S.C. 102(b) as being anticipated by Scales et al. (USPN: 4,914,573) (hereinafter Scales).

As per claims 17 and 18, Scales discloses the invention as claimed including a method of processing a data request within a processing agent comprising posting the data request internally within the agent, determining whether the cache hit by comparing address information of the data request with tags stored in the internal cache, and if cache miss occurs, posting a sequence of

external transactions to fill a cache line with data associated with the data request (e.g. see column 1, lines 12-32); wherein each cache line is sized to store multiple data line lengths of data is taught by Scales since Scales clearly teaches the cache line supports multiple entries having different size (***If the size of the entries in the cache line is different***) (e.g. see column 7, lines 59-62);

As per claim 19; reading the cache coherency state information associated with the requested data when address information matches a stored tag, and identifying cache miss when coherency information is invalid is embedded in Scales's system and being taught to the extent that it is claimed, since Scales discloses in a conventional system the cache are arranged as a plurality of lines, each containing plurality of entries which share a common "tag address", ***and a corresponding number of valid bits which when set indicate the validity of the respective entries in the particular cache line*** (e.g. see Scales's column 1, lines 12-17);

As per claims 20 and 21; Scales teaches the determining whether the request hits a tag stored in the cache by comparing the requested data address information with a tag address and generating a single transaction to read the requested data into the agent (e.g. see column 1, lines 19-21;);

As per claim 23, Scales discloses the invention as claimed

including a method of processing a data request within a processing agent comprising posting the data request internally within the agent, determining whether the request hit in the cache, when cache miss occurs, posting a series of external transaction to fill a cache line with data associated with the data request (e.g. see column 1, lines 12-32); the external transactions directed to a data-line sized data item identified by an address of the data request and to at least one other data-line-sized item adjacent to the first data item (e.g. see column 2, lines 55-66).

As per claims 27 and 28, wherein data line corresponds to the maximum amount of data that can be transferred in a single bus transaction is taught by Sachs to the extent that it is being claimed; for example, when detailing data cache bus, Sachs discloses on store operations, the CPU puts an address following by data on the address/data bus for one (single) clock/bus cycle (e.g. see column 6, lines 27 et seq.);

Rejections - 35 USC § 103

4. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at

> the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

5. Claims 1-7, 11-16 and 25-26 are rejected under 35 U.S.C. 103(a) as being unpatentable over Sachs et al. (USPN: 4,884,197); hereinafter Sachs, in view of Scales, III et al., (USPN: 4,914,573) (hereinafter Scales).

As per claim 1, Sachs teaches the invention as claimed including a processing agent (e.g. see figures 8 and 9) to transfer data of predetermined data length in an external transaction, the agent comprising an cache memory 320 having a plurality of cache entries; for example, Sachs discloses that the cache memory 320 having three fields, a used bit field, and two identical read-write memory fields W and X wherein each field W and X contain multiple lines, and each line in the cache contains multiple entries (fields) (e.g. see figure 10.B) and having tag field RA, line valid bit LV, line dirty bit LD and multiple data entries DT (e.g. see column 22, lines 14 et seq.). Sachs discloses the invention as claimed; however Sachs does not particularly teach that each entry sized to store multiple data line lengths of data. Scales, in his teaching of bus master which selectively attempts to fill complete entries in a cache line, disclose the missing element that known to be required in Sachs in order to arrive at Applicant's current invention wherein Sachs teaches cache memory 22 having a plurality of cache entries

(e.g. see column 2, lines 38 et seq.), each of the cache entries include a tag portion for storing address information RA (e.g. see figure 2, column 2, lines 38 et seq.), cache coherency state field [A3, A2] wherein each entry can be sized to store multiple data line lengths of data (e.g. see column 7, lines 59 et seq.).

Accordingly, it would have been obvious to one having ordinary skill in the art at the time the current invention was made to implement the cache having multiple cache lines comprising multiple entries, wherein each entry sized to store multiple data line lengths of data as being taught by Scales for that of Sachs. In doing so, it would increase the flexibility of Sachs system by allowing it to serve a broader range of applications and their variants thereby broadening one's potential market and saving investment capital.

As per claim 2, Sachs clearly discloses that the cache entries include a tag portion for storing address information as being the real address field RA (e.g. see figure 10B, and column 22, lines 32 et seq.);

As per claim 3, the match detection logic for the tag portions and control logic provided in communication with the match detection logic is taught by Sachs as the comparators 332 and 334, and the multiplexer 341 (e.g. see figure 9, column 21, lines 4 et seq.; for example, Sachs discloses the match/no match signals output from comparators 332 and 334 indicate a cache hit

when the requested real address was presented in the cache and the data was valid, or a cache miss when the requested data is not present in the cache (e.g. see column 21, lines 20 et seq.);

As per claim 4, Sachs clearly teaches each cache line further have a cache coherency state field such that: a cache line valid bit LV, a line dirty bit DT (e.g. see figure 10B, column 22, lines 28 et seq.);

As per claim 5, Sachs discloses his processing agent further comprising a transaction queue as being equivalent to the TLB 350 having a plurality of queue entries (lines) (e.g. see column 22, line 54 et seq.), wherein the queue entries including a primary entry for storing address information and status information of a first external transaction, and a secondary entry for storing status information of a second external transaction is explicitly taught by Sachs as each line having virtual address field VA for storing address information and a reference bit R field or a dirty bit D field for storing status information of an external transaction being read/write transactions (note also an user valid bit UV field, a supervisor valid bit field, a protection level word PL field and a system tag ST field; e.g. see column 22, lines 59 et seq.; column 24, lines 27 et seq.);

As per claim 6, wherein the status information of the first external transaction includes a field (e.g. the reference bit R or the dirty bit D) indicating that the line has been referenced

by a read or write transaction (multiple transaction sequence), or having been modified by a write transactions (e.g. see column 22, lines 59 et seq.; column 24, lines 27 et seq.);

As per claim 7, Sachs discloses the invention as claimed; however Sachs does not particularly teach that the total number of primary and secondary entries equals to the number of data line lengths provided in the cache entries. First of all, it should be noted that the total number of lines in both cache and TLB being disclosed in Sachs's system is a system dependent feature, it can be varied dependent on what system they are implemented within. Secondly, Sachs clearly discloses the W and X memories in BOTH cache memory 320 and TLB 350 each contain multiple lines, and as an example for illustration in the current invention, Sachs selects the number of cache lines being equal to 128 lines (e.g. see column 22, line 22); and the number of lines in the TLB is 64 lines (e.g. see column 22, line 54). Accordingly, it would have been obvious to one having ordinary skill in the art at the time the current invention was made to implement both cache and TLB to have the same number of lines since the numbers of lines in both cache and TLB are changeable as indicated by Sachs. In addition, doing so, it would allow the TLB to buffer more data for reference which results to increasing data hit rate in both TLB and cache, therefore being advantageous.

As per claim 11, Sachs disclose a processing agent (e.g. see figures 8 and 9) comprising an cache memory 320 having a plurality of cache entries; for example, Sachs discloses that the cache memory 320 having three fields, a used bit field, and two identical read-write memory fields W and X wherein each field W and X contain multiple lines, and each line in the cache contains multiple entries (fields) (e.g. see figure 10.B) and having tag field RA, line valid bit LV, line dirty bit LD and multiple data entries DT (e.g. see column 22, lines 14 et seq.). A transaction queue system as being equivalent to the TLB 350 having a plurality of queue entries (lines) to post external transactions, each external transaction related to a single data line (e.g. see column 22, line 54 et seq.), wherein the internal cache and the transaction queue system each receive data requests on a common input (e.g. see figure 8 show common input line being connected to both TLB 270 and cache 220). Sachs discloses the invention as claimed; however Sachs does not particularly teach that each entry sized to store multiple data line lengths of data. Scales, in his teaching of bus master which selectively attempts to fill complete entries in a cache line, disclose the missing element that known to be required in Sachs in order to arrive at Applicant's current invention wherein Sachs teaches cache memory 22 having a plurality of cache entries (e.g. see column 2, lines 38 et seq.), each of the cache entries include a tag portion for

storing address information RA (e.g. see figure 2, column 2, lines 38 et seq.), cache coherency state field [A3, A2] wherein each entry can be sized to store multiple data line lengths of data (e.g. see column 7, lines 59 et seq.). Accordingly, it would have been obvious to one having ordinary skill in the art at the time the current invention was made to implement the cache having multiple cache lines comprising multiple entries, wherein each entry sized to store multiple data line lengths of data as being taught by Scales for that of Sachs. In doing so, it would increase the flexibility of Sachs system by allowing it to serve a broader range of applications and their variants thereby broadening one's potential market and saving investment capital.

As per claim 12, wherein the internal cache and the transaction queue system communicate by signal lines (e.g. see figure 8);

As per claim 13, wherein the signals line include a cache hit signal line and a tag hit signal line (e.g. see figure 9, column 19, lines 58 et seq.; column 20, lines 4 et seq.);

As per claim 14, Sachs teaches the invention as claimed including a processing agent comprising a transaction queue as being equivalent to the TLB 350 having a plurality of queue entries (lines) (e.g. see column 22, line 54 et seq.), wherein the queue entries further comprising a primary sub entry including an address information and status information provided

for a first external transaction, and a secondary subentry provided including a status portion provided for a second external transaction is explicitly taught by Sachs as each line having virtual address field VA for storing address information and a reference bit R field or a dirty bit D field for storing status information of first external transaction being read/write transactions (with respect to status for the second external transaction, note also an user valid bit UV field, a superior valid bit field, a protection level word PL field and a system tag ST field; e.g. see column 22, lines 59 et seq.; column 24, lines 27 et seq.);

As per claim 15, wherein the status portion of the primary entry includes a field representing whether the first transaction is part of a multiple transaction queue is equivalently taught by Sachs as the reference bit R or the dirty bit D field of each line indicating that the line has been referenced by a read or write transaction (multiple transaction sequence), or having been modified by a write transactions (e.g. see column 22, lines 59 et seq.; column 24, lines 27 et seq.);

As per claim 16, Sachs discloses the TLB control logic unit 820 as being equivalent to the control logic being claimed, coupling to the TLB 350 for cycle through the queue entries and post transaction therefrom (e.g. see figure 23);

As per claims 25 and 26, wherein data line corresponds to

the maximum amount of data that can be transferred in a single bus transaction is taught by Sachs to the extent that it is being claimed; for example, when detailing data cache bus, Sachs discloses on store operations, the CPU puts an address following by data on the address/data bus for one (single) clock/bus cycle (e.g. see column 6, lines 27 et seq.);

(10) *New ground of rejection.*

This Examiner's Answer does not contain any new ground of rejection. The rejections of claims 1-7, 11-21 and 23-29 are respectfully maintained. Claims 8-10 and 22 have been withdrawn by Applicant.

(11) *Response to argument.*

As to the remark, Applicant asserted that:

1. "Because claims 17 and 23-24 disclose a many-to-one relationship between data line length and cache line length, these claims are not anticipated by Sachs. For at least the same reasons, claim 29 which depend from claim 24, is not anticipated by Sachs; (e.g. see page 8, lines 23-27).

2. "Because Scales is concerned with cache entries and not cache lines, anticipate the claimed invention." (page 9, second paragraph); Scales fails to teach or suggest a method

of posting a series of external transaction to fill an entire cache line with data; page 9, third paragraph, lines 5 et seq.). Notably, not one of Scales' examples describes a situation in which an entire cache line is filled with data associated with an initial data request (page 10, lines 2 et seq.).

3. Appellants respectfully maintain that even if Sachs were combined with Scales in the manner suggested by the Examiner, the combination would still fail to describe a cache line that is sized to store multiple data line lengths of data. Additionally, because there is no motivation to combine the two references, their combination is not obvious and the Examiner's rejection must be reverse (page 10, third paragraph). The remaining of the arguments also focused to the Applicant's self-defined "many-to-one" configuration as (a) admitted by Examiner that not being taught by Sachs; however (b) clearly disclosed by Scales as being detailed above.

With respect to Applicant's first argument (1), Examiner ***WHOLEHEARTLY*** disagree with Applicant's counsel for the following reasons; for example, the concept of many-to-one (providing cache entries that can store several increments of data length) is irrelevant with respect to claims 24 and 29. For example, first

of all, since claim 24 recites "*a plurality of cache entries, each cache entry sized to store **one** data line length of data*"; secondly, as detailed previously in the Final Office action and above, Sachs discloses all elements that being claimed therein; specifically, with respect to claim 24; Sachs teaches a processing agent (e.g. see figures 8 and 9), the agent comprising an cache memory 320 having a plurality of cache lines; for example, Sachs discloses that the cache memory 320 having three fields, a used bit field, and two identical read-write memory fields W and X wherein each field W and X contain multiple lines, and each line in the cache contains multiple entries (fields) (e.g. see figure 10.B) and having tag field RA, line valid bit LV, line dirty bit LD and multiple data entries DT0-DT4, each cache entry contain multiple bits, e.g. 32 bits (in other words, each cache entry sized to store ***ONE data line length of data*** as being claimed) (e.g. see column 22, lines 14 et seq.); Sachs further discloses that the cache entries include a tag portion for storing address information as being the real address field RA (e.g. see figure 10B, and column 22, lines 32 et seq.); wherein the processing agent posts a series of external transactions related to the address information, each of the external transactions filling one of the cache entries in the cache lines is taught by Sachs as when the requested information for the addressed location is not stored in the respective cache-

MMU memories (cache misses); the micro engine of the cache-MMUs provides a translated physical address for output to the main memory 140, the corresponding information is thereafter transferred from main memory to the respective instruction cache-MMU or to/from the cache-MMU and as needed to the processor 110 (e.g. see column 6, lines 13-21). With respect to claim 29, wherein data line corresponds to the maximum amount of data that can be transferred in a single bus transaction is taught by Sachs; for example, when detailing data cache bus, Sachs discloses on store operations, the CPU puts an address following by data on the address/data bus for one (single) clock/bus cycle (e.g. see column 6, lines 27 et seq.); noting that Sachs illustrated in one of the example (column 22, lines 45-48) that each data word is 32 bits in four-word total data length; Sachs further discloses different modes of ***burst*** data transfer wherein multiple data words (***maximum amount of data, data line***) can be transferred in one cycle or single bus transaction; for example, column 10, lines 30 illustrates the maximum data line (quadword data which corresponds to four-word D0-D4 data length) transfer when cycle type (CT) is in (0,1) mode. For the reasons above, Examiner truly believed Sachs anticipated ***each and every one elements*** being claimed in the claims 24 and 29 of the current invention.

With respect to (2); Examiner would like to point out that

Applicant's counsel wrongly alleged that Scales merely describes the filling of a single cache ***entry*** (not an entire cache *line*). First of all, the many-to-one configuration, as defined by Applicant's counsel, is the configuration in which cache entries can store several increments of data length; (a) this many-to-one configuration (and also the key of the current invention) is clearly taught by Scale (e.g. see column 7, lines 59 et seq., wherein Scales discloses ***if the size of the entries in a cache line is different,*** the size detecting criteria of the state machine 24 must be appropriately adjusted); (b) as known in the art, and logically, in filling the cache line in this many-to-one configuration, each entry of the cache has to be filled one at a time until the entire cache line is filled up; in fact, as taught by Scales, CITED and admitted by Applicant's counsel that "Scales, col. 1:36-38 ("it is an object of the present invention to provide a bus master which selectively attempts to fill entire ***ENTRIES*** in a cache line (noting the plural form). Noting that Scales further clearly teaches this concept, for example, starting on column 1, lines 25 et seq., Scales discloses the filling of entire cache line after filling a full entry; Scales discloses ***after filling a full entry, the bus master then fetches the other operands*** *(posting a series of external transaction)* ***having the same tag address in the system memory to fill the entire cache line.*** In conclusion, it should be noted that the

sole novelty of Applicant's current invention is to support a cache having multiple entries, wherein each entry sized to store multiple data line lengths of data, not to whether filling ***cache entry*** or ***cache line***.

With respect to the last argument (3); Examiner further recognizes that references cannot be arbitrarily combined and that there must be some reasons why one skilled in the art would be motivated to make the proposed combination of primary and secondary references. In re Nomiya, 184 USPQ 607 (CCPA 1975). However, there is no requirement that a motivation to make the modification be expressly articulated. The test for combining references is what the combination of disclosures taken as a whole would suggest to one of ordinary skill in the art. In re McLaughlin, 170 USPQ 209 (CCPA 1971). Sachs et al. and Scales III et al. references are evaluated by what they suggest to one versed in the art, rather than by their specific disclosures. In re Bozek, 163 USPQ 545 (CCPA) 1969. In this case, (a) Sachs and Scales references are in the same field of endeavor of addressing cache memory, (b) the Scales reference which relied on by Examiner for the utilization of multiple data line lengths of data for cache entry wherein Scales teaches cache memory 22 having a plurality of cache entries wherein each entry can be sized to store multiple data line lengths of data (e.g. see figure 2; column 2, lines 38 et seq.; also column 7, lines 59 et

seq.). In doing so, it would increase the flexibility of Sachs system by allowing it to serve a broader range of applications and their variants; therefore being advantageous. Therefore, the 103 rejection in combining Sachs et al. and Scales III et al. references is deemed to be proper.

For the above reasons, it is believed that the rejections should be sustained.

/*TVT*

February 05, 2003

Respectfully submitted,

Tuan V. Thai

PRIMARY EXAMINER

Group 2100

Appeal Conferees:

Peikari Behzad

PRIMARY EXAMINER

Group 2100

Matthew M. Kim

SUPERVISOR PATENT EXAMINER

Group 2100

Clyde E. Findley
KENYON & KENYON
1500 K Street, NW
WASHINGTON, DC 20005
(202) 220-4200